Traffic & Transportation

# IALight: Importance-Aware Multi-Agent Reinforcement Learning for Arterial Traffic Cooperative Control

Lu WEI[1], Xiaoyan ZHANG[2], Lijun FAN[3], Lei GAO[4], Jian YANG[5]

Original Scientific Paper
Submitted: 20 Mar 2024
Accepted: 8 July 2024

[1] Corresponding author, weilu@bgy.edu.cn, Beijing Polytechnic College, School of Information Engineering
[2] zhangxy@bgy.edu.cn, Beijing Polytechnic College, School of Information Engineering
[3] flj@bgy.edu.cn, Beijing Polytechnic College, School of Information Engineering
[4] gaolei@ncut.edu.cn, North China University of Technology, School of Computer Science and Technology
[5] yangj200045@163.com, North China University of Technology, School of Computer Science and Technology



Publisher:
Faculty of Transport and Traffic Sciences, University of Zagreb

**ABSTRACT**
Multi-intersection cooperative control for arterial or network scenarios is a crucial issue in urban traffic management. Multi-agent reinforcement learning (MARL) has been recognised as an efficient solution and shows outperformed results. However, most existing MARL-based methods treat all intersections equally, overlooking their varying importance, such as high traffic volume, connecting multiple main roads, serving as entry or exit point for highways or commercial areas, etc. Besides, learning efficiency and practicality remain challenges. To address these issues, this paper proposes a novel importance-aware MARL-based method named IALight for traffic optimisation control. First, a normalised traffic pressure is introduced to ensure our state and reward design can accurately reflect the status of intersection traffic flow. Second, a reward adjustment module is designed to modify the reward based on intersection importance. To enhance practicality and safety for real-world applications, we adopt a green duration optimisation strategy under a cyclic fixed phase sequence. Comprehensive experiments on both synthetic and real-world traffic scenarios demonstrate that the proposed IALight outperforms the traditional and deep reinforcement learning baselines by more than 20.41% and 17.88% in average vehicle travel time, respectively.

**KEYWORDS**
traffic signal control; intersection importance; multi-agent reinforcement learning; arterial cooperative control.

## 1. INTRODUCTION

Urban traffic signal control (TSC) plays a crucial role in ensuring safety and alleviating congestion for traffic road networks. From the perspective of control scope, TSC methods can be categorised as isolated control, arterial coordination and large-scale network optimisation control. Arterial traffic coordination control is a fundamental aspect of transportation management systems aimed at optimising the flow of traffic along major roadways which plays a vital role in enhancing transportation efficiency, reducing congestion and improving safety. In the past few decades, various methods have been studied and implemented, such as MAXBAND, MULTIBAND etc. However, these existing methods are mostly predetermined, which means they are not suitable for time-varying traffic demand of the arterial intersections, such as variation of the turning movement demand, existing queue and so on.

Reinforcement learning (RL) has emerged as a promising approach for urban traffic optimisation control. Unlike traditional rule-based or pre-programmed systems, RL-based TSC enables learning from interaction experiences in a data-driven manner. Mao et al. [1] introduced a simulation platform to evaluate seven deep

reinforcement learning (DRL) algorithms for isolated intersection control. Their testing results are helpful and shed light on how to select DRL algorithms for various traffic scenarios. For multiple intersections control, multi-agent RL (MARL) was a powerful solution [2]. MARL-based TSC methods can be categorised into three main approaches: centralised, distributed and cooperative methods. The centralised approach trains a global agent for all intersections [3]. However, the main challenge of global agent is that the state size will grow exponentially as the number of intersections increases, making it inefficient for large-scale road network control. The distributed approach treats each intersection as an individual agent which performs an action based on its own local observation [4]. Despite the promising results so far in DRL-based TSC solutions, there are still many major challenges to address before the proposed research can yield real-world products. First, agent formulation including state representation and reward design is a crucial point in DRL-based TSC methods, and inappropriate state and reward design for complex traffic dynamics may lead to slow convergence and unsatisfactory performance measures [5]. Second, most of the DRL solutions for TSC adopt random phase sequence, which is impractical and may severely affect traffic safety. Third, MARL is the primary approach for addressing the cooperative control of multi-intersections in urban road networks. However, to our knowledge, most existing MARL-based TSC methods ignore the importance of each intersection and treat all intersection agents equally.

To address these issues, this paper proposes a DRL-based TSC method named IALight, which optimises the phase duration under a cyclic phase sequence. The main contributions of this paper can be summarised as follows:

1) We propose a novel traffic intensity calculation method which considers both stopped and moving vehicles to support simplified and effective traffic state representation and reward design.
2) We propose a phase green duration optimisation strategy under a cyclic phase sequence to enhance the practicability.
3) We propose a MARL-based method which considers the importance of intersections, such as heavy traffic volume, connecting multiple main roads, etc. A reward refine module which aggregates the original reward of each agent through weighted summation, with the weights being the importance of each intersection.
4) We established a traffic simulation platform based on SUMO. Comprehensive experimental results demonstrate that our IALight method outperforms traditional and baseline RL-based TSC methods.

To the best of our knowledge, our work is the first attempt to consider the importance of intersections in MARL-based TSC methods. Integrating intersection importance in the learning process ensures that the critical intersections, which are located in major traffic corridors or densely populated areas, receive more attention and resources during the decision-making process. Consequently, traffic control can be more targeted and effective.

## 2. RELATED WORK

Traditional TSC methods can be categorised into three types [6]: fixed time, vehicle actuated and adaptive traffic control (ATC). As the most advanced control method in these traditional approaches, ATC systems have been well developed and implemented around the world, such as SCATS [7] and SCOOT [8]. However, the existing ATC methods are driven by theoretical models under some strong assumptions. As we know, it is quite a challenging work to accurately model the complex, dynamic and stochastic nature of the urban traffic system. Therefore, the existing ATC methods may have limited applicability and yield sub-optimal results in real-world applications.

The recent advancement of the RL-based TSC have gained more and more attention. RL agents learn the optimal control program from interactions between agents and road network environments. RL-based TSCs help researchers to improve TSC performance in a data-driven manner, learning from experiences between the agent and environment. Based on the action definition schemes, most existing RL-based TSCs can be classified as phase selection and phase duration optimisation. Phase selection is a discrete action scheme which chooses the next green phase based on traffic state representation and the current policy. DQN and its variants are suitable for these discrete TSC actions. For example, Li et al. [9] take queue length as input and adopt deep stack auto-encoders to calculate the optimal TSC actions. Genders et al. [10] employed a discrete traffic state code (DTSE) to divide the intersection area into grids and calculated the vehicle presence state and average speed within each grid to describe the real-time traffic state of the intersection. They utilised DQN to select the optimal phase from candidate phases. To address the issue of overestimation in DQN, Liang et al [11]

proposed an improvement to the DQN model by using the Duelling Network architecture in deep Q-learning. Compared to the traditional DQN method, this approach significantly enhances the convergence speed of the model. However, these methods allow choosing phases randomly, resulting in irregular and unpredictable signal patterns which may create driver's confusion and lead to potential accident risks in real-world scenarios [12].

Improving learning efficiency is also crucial for enhancing practicality of DRL-based TSC methods. Inappropriate state representation and reward design make the existing DRL-based TSC methods suffer from slow convergence [5]. Recently, some researchers attempted to search support from transportation theory and applications. Max-Pressure (MP) has been recognised as an effective TSC method and it has been theoretically proven to maximise the throughput. PressLight [13] adopted pressure calculation in its RL agent formulation and optimised the control policies to minimise the pressure of intersections. However, PressLight does not consider important vehicle dynamics and the fixed green duration pattern limited its performance. Zhao et al. [5] improved pressure calculation and introduced a new concept of 'traffic intensity' to support element design of their RL agent. Their method was called IPDALight, which combines both intensity and variable phase duration to ensure convergence during model training. However, IPDALight simply includes all vehicles on the road segment in the calculation of traffic intensity. Additionally, IPDALight discretises the signal timing duration in the control, which may not be conducive to accurate and optimal signal timing.

For multiple intersections control in the road network, MARL is a typical and powerful solution. Li et al. [14] proposed a multi-intersection control method based on deep deterministic policy gradient (DDPG). Centralised training and distributed execution (CTDE) training strategy is employed to improve the performance. The comparison results showed that the proposed method can significantly reduce the average waiting time of vehicles. Ma et al. [15] proposed a multi-agent cooperative optimisation method for arterial coordination. The advantages of distributed and centralised learning are integrated to balance global measures and algorithm efficiency. Haddad et al. [2] studied a novel cooperative MARL method for multi-intersection control. To improve the overall control effectiveness of the road network, they designed a mechanism which sharing decisions and observations among agents. Wei et al. [10] proposed a multi-intersection coordination algorithm named PressLight. By considering the impact of neighbouring intersections, PressLight optimises signal timings globally to achieve smoother traffic progression, reduce travel time and enhance overall network performance. CoLight [16], as a current state-of-the-art TSC method for multi-intersection scenario, introduce graph attention on observations to achieve cooperative control. Chen et al. [17] designed RL agents for large-scale network control inspired by Max-Pressure (MP) traffic control method. Comprehensive experiments including a real-world scenario with 2510 traffic lights in Manhattan were conducted to show the performance and generalisation ability of their method. However, these MARL-based methods treat each agent equally, ignoring the importance of intersections in the road network. Some recent research studies have revealed that traffic congestion manifests a 'cascading failures' phenomenon in road networks [18], indicating that the intersections at the source of congestion play important roles in traffic management and control. Addressing this issue, Xu et al [19] proposed an optimisation control strategy for critical intersections in road networks. They first constructed a three-partite graph model of the road network and used a data-driven method to identify these critical intersections. Then, DRQN model was adopted to learn the optimal control strategy for critical intersections. However, the influences between critical intersections and other intersections were neglected in their method, failing to guarantee global optimality at the regional level.

## 3. PROBLEM STATEMENT

Arterial traffic coordination control is a typical traffic scenario which refers to management and optimisation of traffic flow on arterial roads or corridors. The primary objective of arterial TSC is reduce travel time or stop delay for most vehicles by coordinate traffic signals along the arterial. Take *Figure 1* for a simple illustration, the arterial is a control sub region $R_1$ which consists of three intersection agents, denoted as $J_1, J_2, J_3$, represents the set of all signalised intersections. Each intersection has four entry-exit sides, where each side consist of three lanes. However, the traffic condition and location in the network of each intersection determine their varying importance for arterial coordination control. The intersection which holds the highest importance is commonly referred to as a critical intersection, as $J_2$ intersection in *Figure 1*.

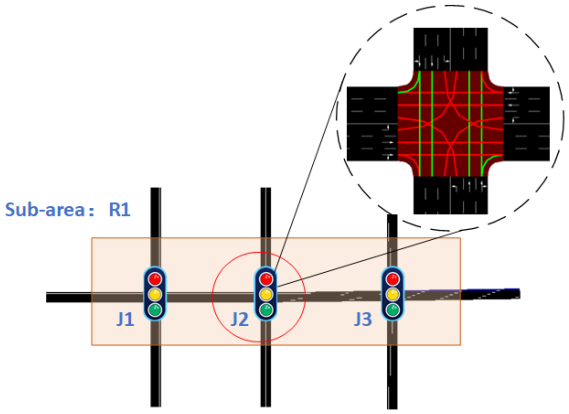


*Figure 1 – Description of control scenario*

We modelled the arterial TSC problem as a decentralised partially-observed Markov Decision Process (Dec-POMDP) which defined by a tuple of $< \mathrm{N}, \mathrm{S}, \{\mathrm{O}^i\}_{i\in\mathrm{N}}, \{\mathrm{A}^i\}_{i\in\mathrm{N}}, \mathrm{P}, \{\mathrm{R}^i\}_{i\in\mathrm{N}}, \gamma >$, Here, $\mathrm{N}$ is the number of agents, $\mathrm{S}$ is the global state space of the entire traffic environment. $\mathrm{O}^i$ is the local observation of agent $i$, $\mathrm{A}^i$ is the action space of agent $i$, $\mathrm{P}: \mathrm{S}\times\mathrm{A}^1\times\mathrm{A}^2 ... \mathrm{A}^{\mathrm{N}} \to \mathrm{S}'$ denotes the state transition probability, $\mathrm{R}^i: \mathrm{S}\times\mathrm{A}^i \to \check{}$ indicates the immediate reward of agent $i$, $\gamma \in [0,1]$ is the discount factor.

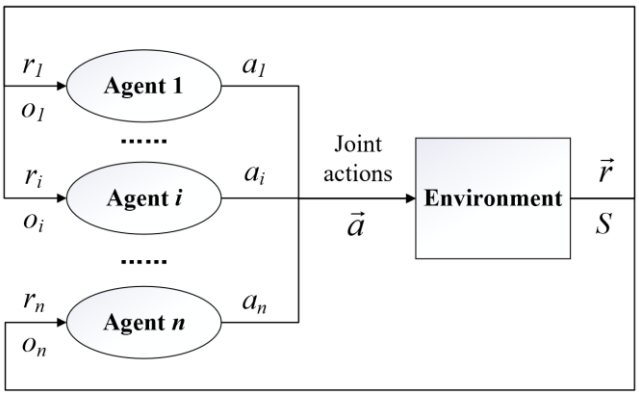


*Figure 2 – The interaction process of MARL*

The interaction between agents and traffic environments is shown in *Figure 2*. At each time step $t$, each agent select its action $a_t^i \in \mathrm{A}^i$ according to the local observation $o_t^i$ and the decentralised policy $\pi_{\theta_i}(a_t^i \mid o_t^i)$. All actions form a combined action $\vec{a}_t = \{a_t^i\}_{i=1}^{\mathrm{N}}$ and can be executed. After this, the traffic state of the arterial transfers from $s_t$ to $s_{t+1}$ according to $\mathrm{P}$. Subsequently, reward $r_t^i$ was received by each agent from the environment.

$$J = \mathrm{E}\left[\sum_t^T \gamma^t \sum_i \alpha_i r_t^i\right] \tag{1}$$

where $r_i$ denotes the scalar reward of agent $t$, $\alpha_i$ is the importance weight of agent $i$.

## 4. METHODOLOGY

In this section, we first present the agent formulation details. Then, the important-aware MARL design is described and discussed. Each intersection is modelled as a DDPG agent with actor-critic framework with centralised training and distributed execute (CTDE) paradigm. The agent formulation including state representation, action definition and reward design will be presented first. Then our IALight model improved from multi-agent deep deterministic policy gradient (MADDPG) is introduced.

### 4.1 Agent formulation

*Observation and state representation*

The observation representation needs to quantitatively reflect the traffic flow condition around the intersections. Queue length, traffic volume, speed and signal phase status are commonly used indicators of traffic conditions. Mao et al. [1] recommended queue length as a reference indicator for observation design based on a series of simulation experiments. However, queue length cannot fully express the traffic demand for traffic signals. For example, at a signalised intersection, there may be a scenario where there are no queues, but a dense vehicle platoon is approaching and will reach the intersection within a few seconds. In this case, the queue length is zero, but the actual traffic demand is significant. Therefore, inspired by MonitorLight [22], we divide the improved lane traffic pressure into two parts: static and dynamic. For lane $i$, the improved pressure can be expressed as follows:

$$P_i = q_i + \sum d_{i,p} \times N_p \tag{2}$$

where $q_i$ denotes as the queue length on lane $i$, which represents the static pressure. $\sum d_{i,p} \times N_p$ denotes the dynamic pressure which is calculated by the density of the predict platoon $d_{i,p}$ multiplied by the platoon length $N_p$.

Furthermore, the impact of the traffic light status and timings are also considered in the observation representation. At time $t$, the observation of each agent can be expressed as follows:

$$o_t = \{P_1, ..., P_j, ..., P_M, \xi_t, \tau_1, ..., \tau_j, ..., \tau_K\} \tag{3}$$

where $P_j$ denotes the pressure of lane $j$, $M$ is all lanes at a single intersection, $\xi_t$ is an One-Hot encoding vector of current traffic signal phase status with 1 denoting green and otherwise 0. $\tau_j$ is the phase timing of phase $j$, $K$ is the number of phases.

*Action definition*

The action pattern can be classified into two main categories: phase selection with unfixed sequence and phase duration calculation with fixed sequence. The former is to decide whether to keep the current phase or switch to a phased which randomly choosing from alternative phases. However, variable phase order may make drivers confused and potentially suffers from traffic safety risks in specific scenarios. The latter attempts to adjust the phase green duration under a fixed phase sequence. Considering the practicality and reliability, we adopt the latter one suggested by [21, 23, 24, 25]. The definition of the intelligent agent actions is as follows:

$$g_t = F(\pi(o_t)) \tag{4}$$

where $F(\cdot)$ is a function which converts the agent action value to phase timing. The traffic control logic based on agent action is shown in *Figure 3*.

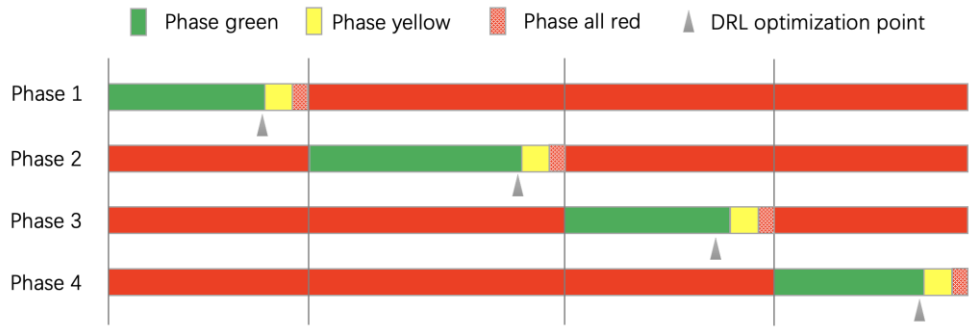


*Figure 3 – The traffic control logic of each agent*

The grey triangle represents DRL optimisation decision points at which the DRL model will compute optimal green time for next phase based on current state and reward. To ensure clearance traffic safety, the yellow and all red time are added to the green end of each phase. Furthermore, it is necessary to impose the following constraints on the actions based on the minimum green and maximum green signal duration to provide stable and reliable control:

$$g_{\min} \le g_t \le g_{max} \tag{5}$$

where $g_{\min}, g_{\max}$ denotes the minimum green and maximum green time constraints for safety and fairness, respectively.

*Reward design*

The reward $r_t^i$ received by agent $i$ at time step $t$ is a scalar feedback signal after action $a_t^i$ is taken. It is a fundamental concept used to guide the learning process of the agent. The design of the reward function is crucial in reinforcement learning, as it determines the agent's learning objectives and influences its behaviour. The main objective of our method is to improve the control performance of arterial traffic. A common metric used to indicate traffic efficiency is the total vehicle travel time. However, using the total travel time as feedback to the model may lead to delayed reward, which is unreasonable [20]. In the study by Wei et al. [13], it was verified that reducing the pressure at intersections is equivalent to reducing the average travel time. Therefore, we define the reward of agent $i$ as the sum of the pressure on all incoming lanes, which is expressed as follows:

$$r_t^i = -\sum_{j=1}^{M} P_{j,t} \tag{6}$$

where $M$ is all lanes of intersection $i$, $j$ is the lane index. The negative sign indicates that minimising the reward function corresponds to minimising the intersection's pressure, which aligns with the goal of reducing congestion and improving traffic flow efficiency.

## 4.2 The proposed IALight method

In this section, we introduce our IALight method, which considers importance of intersection agents based on MADDPG. A reward adjustment module (RAD) is designed to incorporate agent importance in the MARL framework and model training process, as shown in *Figure 4*.

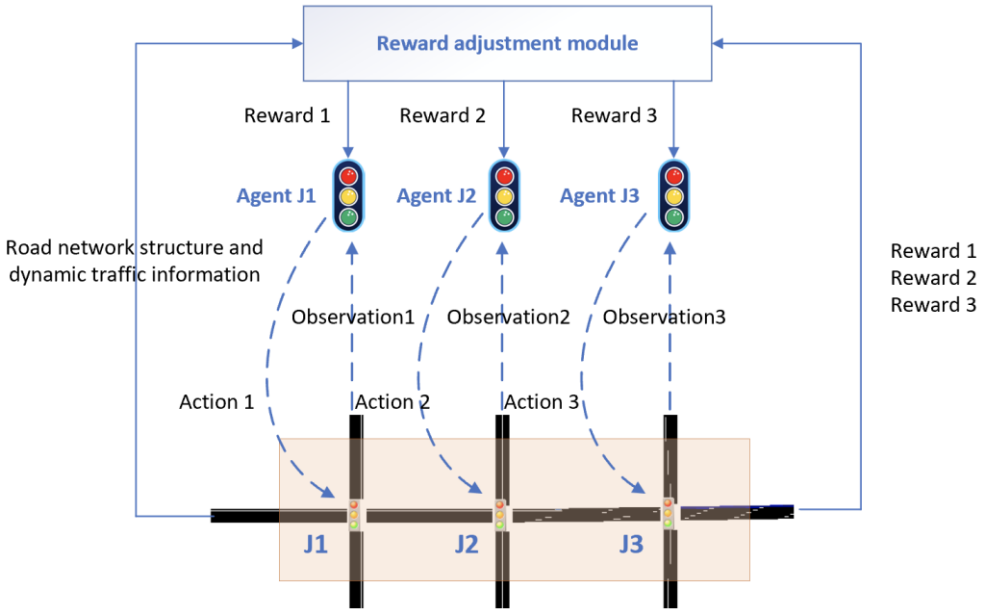


*Figure 4 – The IALight framework*

The instantaneous rewards obtained by each intersection agent are sent to the RAD. This module adjusts the reward signals based on the importance of each agent and the global control objective. The importance evaluation method can be seen in [26, 27]. The output of RAD is a modified reward which contains importance and cooperative target information. The modified reward for training can be expressed as follows:

$$r_t^{i'} = \sum_i \alpha_i r_t^i \tag{7}$$

where $r_t^i$ denotes the reward received by agent $i$ at time $t$, $\alpha_i$ is the importance factor of agent $i$, $0 \le \alpha_i \le 1$ and $\sum_i \alpha_i = 1$.

Each intersection of the arterial is modelled as an agent with actor and critic network, as shown in *Figure 5*. The critic network takes the joint observations and actions of all agents as input and estimates the value function, which represents the expected cumulative return. The actor network aims to maximise this value by adjusting its policy, while the critic network is used to provide feedback on the quality of the chosen actions. One of the key challenges in multiagent settings is the non-stationarity problem, where the environment dynamics change as agents learn and update their policies. To address this issue, the MADDPG utilises a replay buffer, similar to the DDPG, which stores past experiences of the agents. During training, the agent's sample a minibatch of experiences from the replay buffer. The minibatch is used to update both the actor and critic networks. The critic network is updated by minimizing the difference between the predicted Q-value and the target Q-value. The actor network is updated using feedback from the critic network, with the goal of maximizing the Q-values provided by the critic, which corresponds to selecting actions that yield higher rewards. Both the actor and critic networks have target networks, which are updated gradually over time using a soft update mechanism. To balance exploration and exploitation, a noise process is added to the actions taken by the actor, encouraging the agent to explore a variety of actions in the environment.

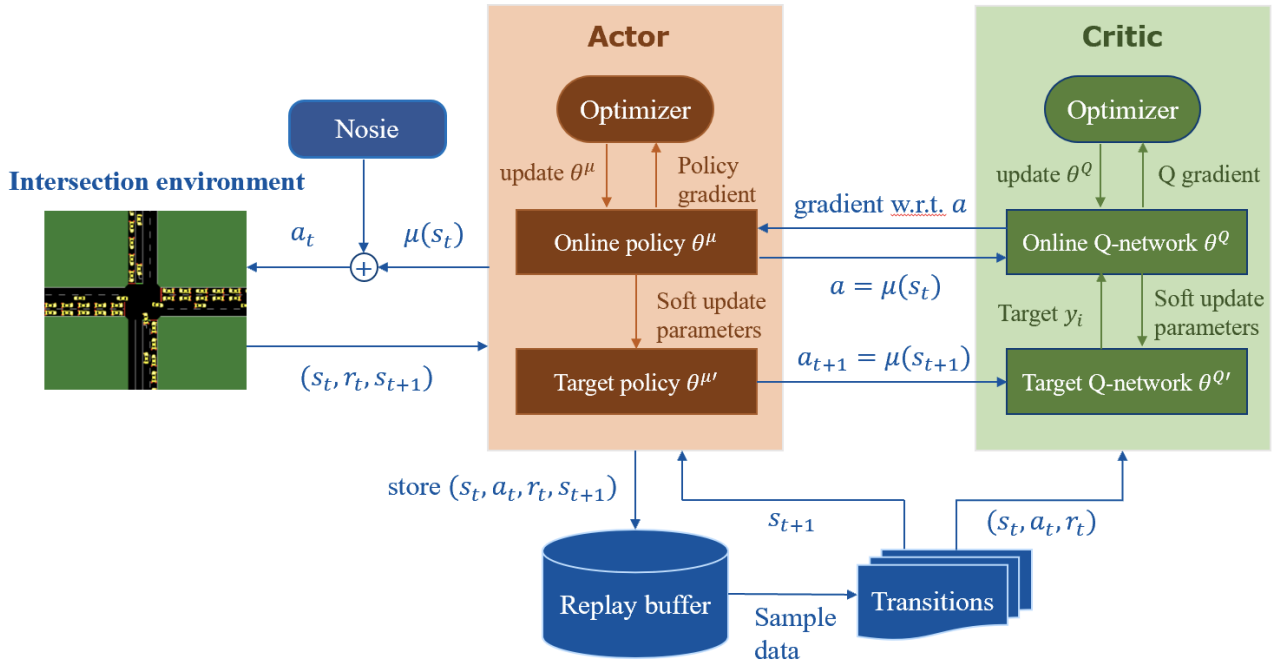


*Figure 5 – The network of each agent*

### 4.3 Model training

In this section, we introduce model training of our IALight. Similar with the MADDPG, we employ the centralised learning and decentralised execution (CTDE) approach [28] for model training, as shown in *Figure 6*.

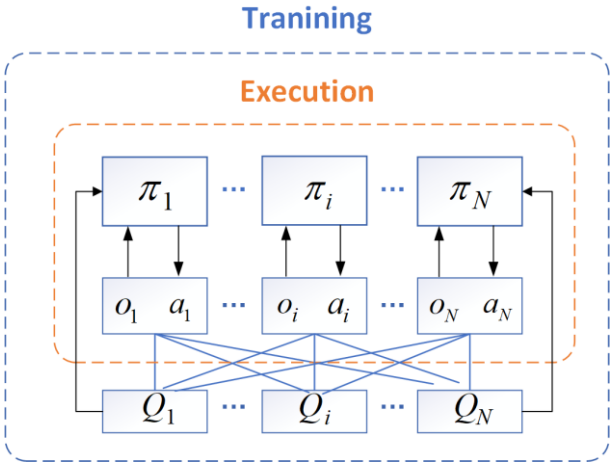


*Figure 6 – Overview of the CTDE training*

During centralised training, the critic network utilised the joint observations and actions of all agents as input, enabling the agents to learn a global perspective of the environment. On the other hand, during execution, each agent only utilises its local observation and the learned actor network to choose actions, thereby ensuring decentralised deployment.

The policies of agents can be expressed as $\mu = \{\mu^1, \mu^2, ..., \mu^N\}$ with corresponding parameters as $\theta = \{\theta^1, \theta^2, ..., \theta^N\}$. The policy gradient of agent $i$ can be written as follows:

$$\nabla_{\theta^i} J(\mu^i) = \mathrm{E}\left[\nabla_{\theta^i} \mu^i(a^i \mid o^i) \nabla_{a^i} Q^{i,\mu}(x; a^1, ..., a^N)\big|_{a^i=\mu^i(o^i)}\right] \tag{8}$$

where $o^i$ is the observation of agent $i$, $x = [o^1, o^2, ..., o^N]$ indicates global state, $Q^{i,\mu}(x; a^1, ..., a^N)$ is the centralised action-value function which input all agents' actions and state $x$, and then output $Q$ of agent $i$.

The parameter update method of critic is minimised loss function, the loss function is defined as follows:

$$\mathrm{L}(\theta^i) = \mathrm{E}_{x,a,r,x'}\left[(Q^{i,\mu}(x, a^1, a^2, ..., a^N) - y)^2\right] \tag{9}$$

where $y$ is the TD target which can be calculated as follows:

$$y = r^i + \gamma Q^{i',\mu'}(x'; a^{1'}, ..., a^{N'})\big|_{a'=\mu'(o)} \tag{10}$$

where $\mu' = \{\mu'_{\theta^1}, ..., \mu'_{\theta^N}, \}$ denotes the target policy network. In reinforcement learning, the target network and the online network are commonly used to achieve stability and convergence of the learning process. Soft update is a method used to synchronise the parameters between these networks.

Soft update is a progressive updating method that gradually updates the parameters of the target network by smoothly blending them with the parameters of the online network. Specifically, during each update of the target network, only a small portion of the parameters is updated, while the remaining parameters are kept unchanged or updated with a small magnitude. This progressive updating approach allows the parameters of the target network to gradually approach those of the online network, reducing instability and drastic fluctuations during the updating process.

During each update of the target network, compute the blended value of the parameters using the following formula:

$$\theta' = \tau\theta + (1-\tau)\theta' \tag{11}$$

where $\theta'$ represents the parameters of the target network, $\theta$ represents the parameters of the online network, and $\tau$ denotes the soft update rate.

The whole algorithm is summarised in *Table 1*:

*Table 1 – IALight algorithm based on the MADDPG*

**Algorithm 1: IALight based on MADDPG**

Initialise critic networks $Q^i$ and actor network $\mu^i$ with random parameters $\theta^{i,Q}$, $\theta^{i,\mu}$ of each agent $i$.
Initialise target networks $Q'^i$ and $\mu'^i$ with weights $\theta^{i,Q'} \leftarrow \theta^{i,Q}$, $\theta^{i,\mu'} \leftarrow \theta^{i,\mu}$.
Initialise reply buffer $\mathcal{D}$.
**for** episode=1 to M **do**
  Initialise a random process $\mathcal{N}$ for action exploration.
  Receive initial state $x$ and $o^i$ for each agent.
  **for** $t = 1$ to *episode_length* **do**
    for each agent $i$, select action $a^i = \mu_{\theta^i}(o^i) + \mathcal{N}_t$ w.r.t the current policy and exploration.
      execute actions $a^1, \dots, a^N$ and observe reward $r^i$ and new state $x'$.
      calculate $r'$: $r' = \sum_i \alpha_i r^i$.
      store $(x, a, x', r')$ into replay buffer $\mathcal{D}$.
      $x \leftarrow x'$.
      **for** agent $i = 1$ to $N$ **do**
        sample a random batch of transitions $(x_j, a_j, r_j, x'_j)$ from $\mathcal{D}$.
        calculate TD target: $y_j = r_j + \gamma Q'^{i,\mu'}(x'_j; a^{1'}, \dots, a^{N'})|_{a^{k'}=\mu^{k'}(o_j^k)}$.
        update critic by minimise the loss: $\mathcal{L}(\theta^{i,Q}) = \frac{1}{S}\sum_j \left(y_j - Q^{i,\mu}(x_j; a_j^1, \dots, a_j^N)\right)^2$.
        update actor using the sampled policy gradient:

$$\nabla_{\theta^{i,\mu}} J \approx \frac{1}{S}\sum_j \nabla_{\theta^{i,\mu}} \mu^i(o_j^i) \nabla_{a^i} Q^{i,\mu}(x_j; a_j^1, \dots, a_j^N)|_{a^i=\mu^i(o_j^i)}$$

      **end for**
    update target network parameters for each agent $i$:

$$\theta^{i,Q'} \leftarrow \tau\theta^{i,Q} + (1-\tau)\theta^{i,Q'}$$
$$\theta^{i,\mu'} \leftarrow \tau\theta^{i,\mu} + (1-\tau)\theta^{i,\mu'}$$

  **end for**
**end for**

## 5. EXPERIMENTS RESULTS AND DISCUSSION

The simulation experiments are conducted on the Simulation of Urban Mobility (SUMO) [29] simulator with version 1.15.0. Our model is built with Python3 and PyTorch 2.0.1. Data interaction between the simulation and the RL model is achieved through the Traffic Control Interface (TraCI), an open interface provided by SUMO for traffic control. This allows for the retrieval traffic states from SUMO. Also, the traffic control action can be performed in the SUMO road network. The data interaction process between SUMO and DRL models is illustrated in *Figure 7*.

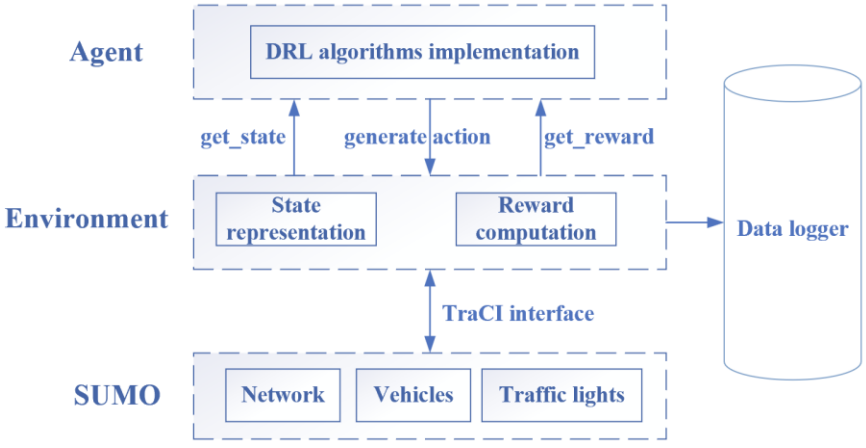


*Figure 7 – Data interaction process*

### 5.1 Simulation settings

We introduced two road networks for model evaluation. The first is a synthetic arterial network consisting of three homogeneous intersections (1×3), as shown in *Figure 8*. Each intersection has four entries with three lanes, respectively. The second is a real-world arterial in Beijing, China, with five heterogeneous intersections as shown in *Figure 9* which was imported from the Open Street Map (OSM).

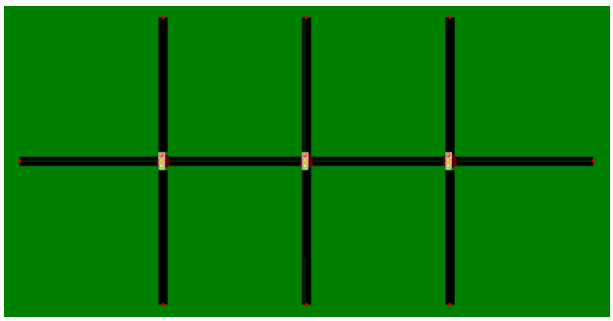

*Figure 8 – 1×3 synthetic road network layout*

*Figure 9 – Road network of Yangzhuang in Beijing*

We first introduced the 1 × 3 road network. Each intersection in the road network follows the same 4-phase signal timing plan. The sine wave as rate parameter for dynamic traffic demand is used to create schedule for number of vehicles to be generate each simulation second. Besides, we randomly shift traffic pattern as a form of data augmentation. The traffic demand data is generated dynamically at every simulation step with an exponential distribution of headway. The importance of each intersection is predetermined as $\alpha_1 = 0.3$, $\alpha_2 = 0.3$, $\alpha_3 = 0.4$. In fact, there have been many research studies and methods proposed to rank or evaluate the intersections in traffic networks, such as complex network theory, graph theory and PageRank etc. For example, Liu et al [32] utilised a graph attention neural network to estimate the importance of each intersection. Then, they introduced the importance into the reward function to find the optimal traffic light scheme. Huang et al [33] proposed a novel traffic node importance evaluation method based on clustering in represented transportation network. Xu et al. [19] identified critical nodes which would cause a dramatic reduction in traffic efficiency of the network if they were fail. Focus on these important nodes, they introduce a novel traffic signal control approach based on deep reinforcement learning. In summary, different important values of intersections imply their different locations in the road network, different traffic volumes they carried or the scope of influence and propagation speed when their congestion occurs. All these factors directly affect the optimisation targets design and outcomes of traffic control algorithms.

## 5.2 Experimental settings

*Hyperparameter settings*

Both the actor network and the critic network are used to feed forward fully connected neural network with two hidden layers, since our state representation is designed as vectors, not image-like. The detail network structure information of the actor and the critic can be seen in *Table 2*.

*Table 2 – Network structure of synthetic arterial scenario*

| Network | | Layers | Dimension | Activation |
|---|---|---|---|---|
| Actor | Input | 1 | 21 | N/A |
| | Hidden | 2 | 63 | Elu |
| | Output | 1 | 1 | Tanh |
| Critic | Input | 1 | 66 | N/A |
| | Hidden | 2 | 198 | Elu |
| | Output | 1 | 1 | N/A |

The hyperparameters of the proposed IALight are shown in *Table 3*, both common DRL hyperparameters and general TSC parameters are included. In common DRL hyperparameters, the discounting factor $\gamma$ is used to adjust the influence of short-term and long-term effects. The learning rates of the actor and the critic determine the size of the steps taken towards the optimal solution respectively. The batch size determines the number of training samples used in each iteration of the training process. The soft update factor $\tau$ controls the rate at which the target network parameters are updated, while the update target frequency determines how often the target network can be updated. The replay frequency determines how often the online network is updated.

*Table 3 – Hyperparameter settings*

| Parameter | Description | Value |
|---|---|---|
| $\alpha_{actor}$ | Actor learning rate | 0.0001 |
| $\alpha_{critic}$ | Critic learning rate | 0.0005 |
| $\gamma$ | Discount factor | 0.99 |
| $N_{batch}$ | Batch size | 32 |
| $N_{buffer}$ | Buffer size | 20000 |
| $\tau$ | Soft update factor | 0.01 |
| $Freq_{replay}$ | Replay frequency | 32 |
| $Frea_{target}$ | Update target frequency | 128 |
| $g_{\min}$ | Minimum green time | 5 |
| $g_{\max}$ | Maximum green limit | 65 |
| $C_{\min}$ | Minimum cycle limit | 40 |
| $C_{\max}$ | Maximum cycle limit | 255 |

For traffic signal parameters, $g_{\min}$ serves an important role in ensuring traffic safety and efficiency at signalised intersections, it provides sufficient time for vehicles to cross an intersection or a traffic flow to

progress smoothly when the green light start. $g_{max}$ is used to minimise total delays and balance the needs of different traffic movements. $C_{\min}$ and $C_{\max}$ are cycle length restraints. We obtain these constraint values from practical knowledge and projects in traffic engineering. In fact, the $g_{\min}$ and $g_{max}$ need to be calculated based on the specific conditions of each intersection, such as vehicle speed, the size of intersection, the length of segments, etc. Taking a four-phase controller intersection as an example, if $g_{\min}$ is set to 5s, the minimum cycle $C_{\min}$ is derived from the sum of minimum green (5s), yellow (2s) and all red (2s) of each phase, i.e. 40 seconds. The value of $C_{\max}$ we presented here is considered with as many traffic signal controllers or standards that use a byte to store the duration of the cycle, and its maximum value is 255.

*Baselines*

We compared our method with two categories of baseline methods: conventional traffic control methods and DRL-based methods. All baseline methods are simply described as follows:

— Self-Organised Traffic Light (SOTL) [30]: A conventional approach which dynamically adapts traffic signal timing and sequencing based on real-time traffic conditions and demand.
— Max-Pressure (MP) [31]: The MP controller is a network-level adaptive control method that has advantages over other traditional methods. Each intersection calculates a pressure based on the queues in adjacent links, then selects the stage with the highest pressure.
— Independent DQN (I-DQN): Each intersection is controlled by a DQN model with fixed green time, there is no communication or information sharing among each DQN agent.
— Independent DDPG (I-DDPG): Each intersection is controlled by a DDPG model with fixed phase sequence and variable green time, there is also no communication or information sharing among each DDPG agent.
— PressLight: A state-of-the-art RL method for TSC which optimises the pressure of each intersection based on DQN.
— IPDALight: An efficient RL-based method that combines the merits of both the newly introduced concept of intensity and variable phase duration.

*Evaluation metrics*

The performance of different methods is evaluated by the following metrics:

— Episode Reward: Average reward of each episode which is used to show the training efficiency and convergence.
— Average queue length (AQL): Average queue length of each simulation step at an intersection, can be calculated as follows:

$$Q_t^{avg} = \frac{1}{N}\sum_{i=1}^{N} Q_t^i \tag{12}$$

where $N$ is the total number of movements at the intersection, $Q_t^i$ is the queue length of movement $i$ at time $t$. A shorter AQL represents fewer cars waiting on all movement entry links.

— Average travel time (ATT): Average travel time of all vehicles at an episode. The travel time of vehicles is calculated by the time between when the vehicle enters the road network and arrives at its destination. The average travel time of an episode can be expressed as follows:

$$T^{avg} = \frac{1}{N}\sum_{i=1}^{N}(t_i^s - t_i^e) \tag{13}$$

where $N$ denotes the total number of vehicles that finish their trips, $t_i^s$ and $t_i^e$ are the departure time and arrival time, respectively. A lower ATT means a better operation efficiency of traffic systems.

— Average vehicle speed (AVS): Average vehicle speed is calculated by dividing the cumulative speed of all vehicles by the total number of vehicles, denoted as follows:

$$V^{avg} = \frac{1}{N}\sum_{i=1}^{N} v_i \tag{14}$$

where $N$ denotes the total number of vehicles, $v_i$ is the speed of vehicle $i$. A higher AVS means a smoother traffic operation.

— Average Waiting Time (AWT): Average waiting time of vehicles at time t or in each episode:

$$W^{avg} = \frac{1}{N}\sum_{i=1}^{N} w_i \tag{15}$$

where $N$ denotes the total number of vehicles, $w_i$ is the waiting time of vehicle $i$. The lower AWT means a higher operation efficiency of intersections along the arterial. The AWT can be treated as delay in some special scenarios.

### 5.3 Convergence comparison

The training convergence comparison results are shown in *Figure 10*. We adopt a sliding average to smooth the series data and reduce data noise. This will help us to observe the trends of the data, without being disturbed by short-term fluctuations. It should be noted that we only presented the training results in the 1×3 synthetic road network. In order to provide a clear convergence comparison, we normalised the episode reward of each DRL-based method because each reward design is different.

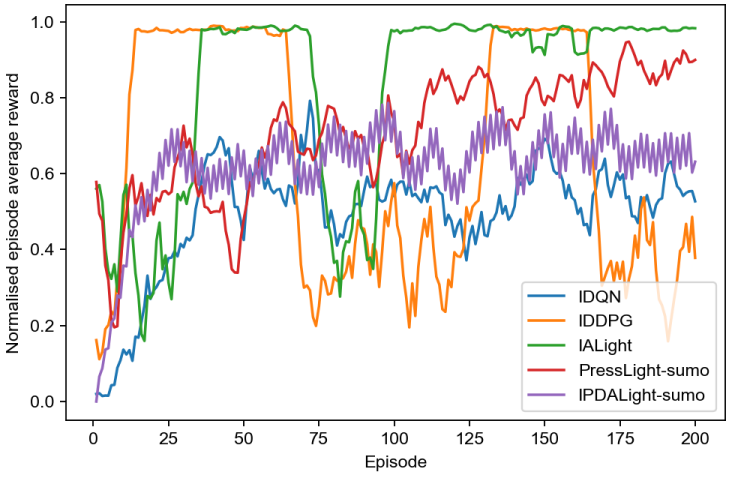


*Figure 10 – Average rewards for each training episode*

*Figure 10* demonstrates that our IALight has a better convergence and higher reward after 100 episodes of training. Although the I-DDPG is capable of quickly achieving high reward values, its performance becomes highly unstable in later episodes. We have rewritten the PressLight and IPDALight methods based on the SUMO simulation platform (the original methods are conducted on CityFlow), the network of agents is set the same as the original method. However, the stability performance of training is not particularly satisfactory.

### 5.4 Performance results and discussion

In this section, we present the evaluation metrics comparisons which defined in Section 5.2 for the proposed IALight methods and all traditional and DRL-based baselines. The overall performance in the 1×3 synthetic road network is shown in *Figure 11, 12 and 13*. The total number of simulation steps for each test episode is 7200, just like the training settings. *Figures 11, 12 and 13* provide a queue, average speed and delay metrics comparison for each intersection in the arterial. From the figures we can see that the IALight has a better performance when it comes to improving the efficiency for intersections along the arterial.

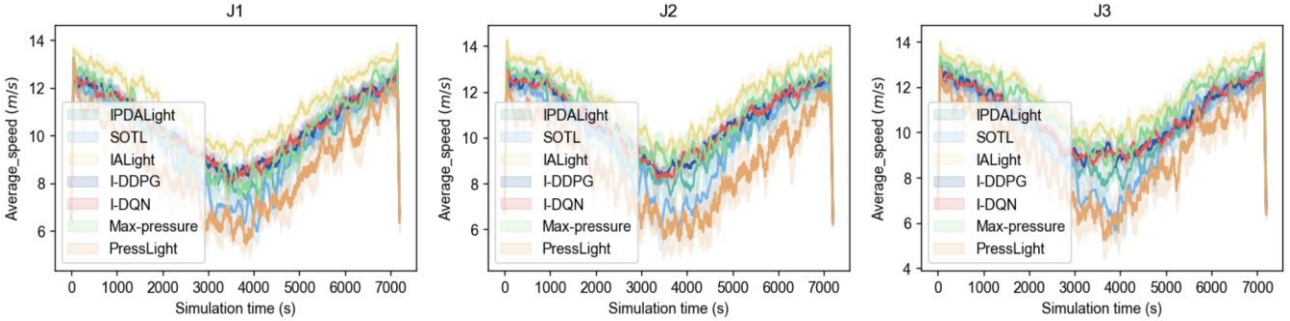


*Figure 11 – Average speed comparison of intersections on the 1×3 synthetic arterial*

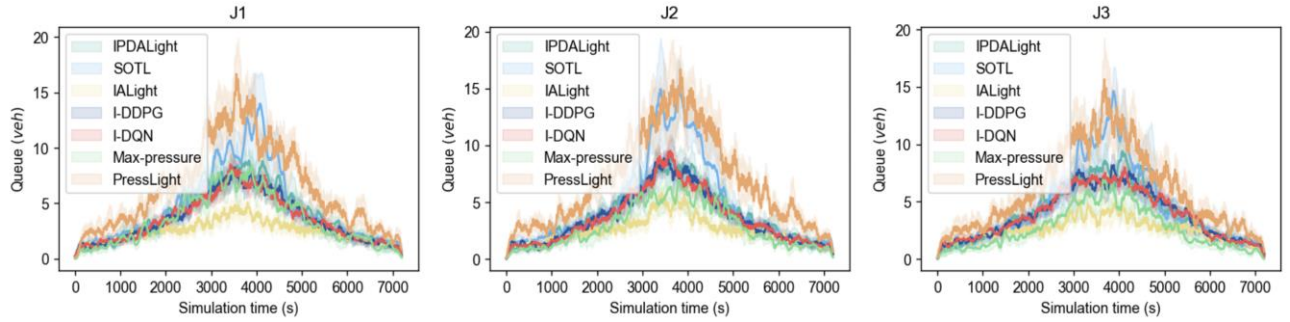


*Figure 12 – Queue comparison of intersections on the 1×3 synthetic arterial*

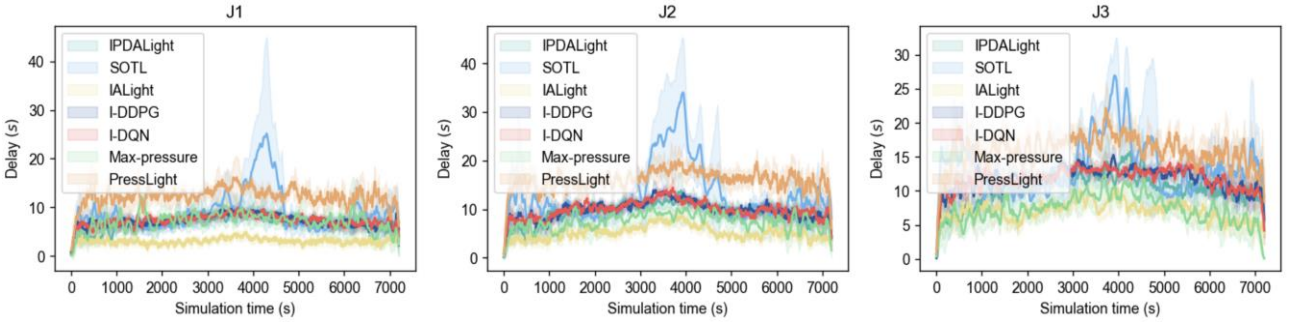


*Figure 13 – Delay comparison of intersections on the 1×3 synthetic arterial*

In order to provide a clearer comparison, we further summarised the arterial metrics including the AQL, ATT and AWT in *Table 4*. The unit for the AQL is vehicle, while the unit for both ATT and AWT are seconds, and for the AVS, the unit is m/s. The DRL-based methods can generally perform better than the conventional methods after well trained and optimised hyperparameters. This demonstrates that the proposed IALight can promote better cooperative control for arterial intersections.

*Table 4 – Metric comparison on 1×3 synthetic arteria*

| Method | AQL [veh] | ATT [s] | AWT [s] | AVS [m/s] |
|---|---|---|---|---|
| SOTL | 7.835 | 88.183 | 11.843 | 8.954 |
| Max-Pressure | 4.110 | 73.906 | 10.059 | 10.234 |
| I-DQN | 4.693 | 77.737 | 11.290 | 9.936 |
| I-DDPG | 3.581 | 62.628 | 5.168 | 10.966 |
| PressLight-sumo | 3.624 | 67.985 | 5.515 | 10.754 |
| IPDALight-sumo | 3.775 | 69.133 | 5.401 | 10.604 |
| **IALight** | **3.570** | **55.018** | **5.005** | **11.642** |

We present a time-space diagram as shown in *Figure 14* to demonstrate the cooperative control effectiveness of our method. The time-space diagram is suggested by the Highway Capacity Manual (HCM) to analyse arterial progression for a set of traffic signal timing plans along the arterial. The x-axis is simulation time, and the y-axis is the distance between intersections. We store the speed and location information for each vehicle in the network at every time step by using the FCDOutput option of the SUMO, and then plot the trajectories of vehicles travelling from west to east and the corresponding traffic signal status duration along the arterial as an example. As demonstrated in the *Figure 14*, most vehicles are able to pass through the following two intersections without additional stops after departing from the first intersection.

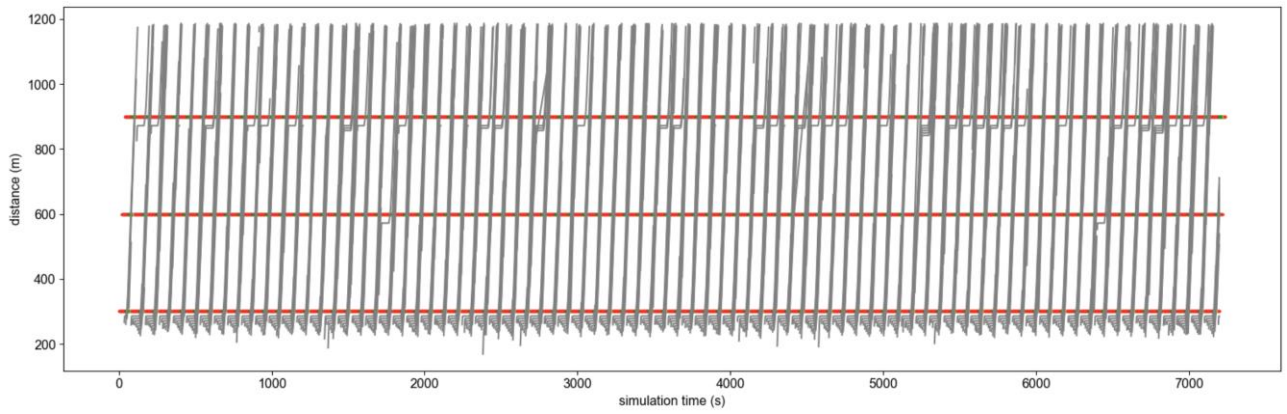


*Figure 14 – Time-distance diagram of the 1×3 synthetic arterial*

In addition to synthetic arterial experiments, we further conducted an experiment on the Yangzhuang Street, which consist of five intersections in Beijing. Unlike the synthetic network, the traffic signal phases and sequence of each intersection in the Yangzhuang Street are heterogeneous, as shown in *Figure 15*. What needs to be noted is that the fourth intersection is a pedestrian crosswalk. However, pedestrian traffic was not considered in the research conducted here. Therefore, we predefined a fixed green time for the pedestrian phase at intersection 4. The dynamic traffic demand is generated based on the traffic flow data collection by geomagnetic sensors installing on each approaching lane.

| Intersection | Traffic signal stages | | | |
|---|---|---|---|---|
| | 1 | 2 | 3 | 4 |
| 1 | ↓ | ↓ ↑ | ← | ↑ |
| 2 | ↖ ↘ | ↓ ↓ ↘ | ↓ ↑ | ← → |
| 3 | ↓ ↑ | ↖ ↘ | ← → | |
| 4 | ↓ ↑ | ↔ ↔ | | |
| 5 | ↓ ↑ | ↖ ↘ | ← → | |

*Figure 15 – Traffic signal phase settings*

*Figure 16* illustrates the result of the overall performance comparison between the IALight and baselines. The x-axis is the evaluation metrics, and the y-axis represents corresponding values for each metric. In *Figure 16*, the proposed IALight can outperform all other benchmark methods on the real-world network's simulation experiment.

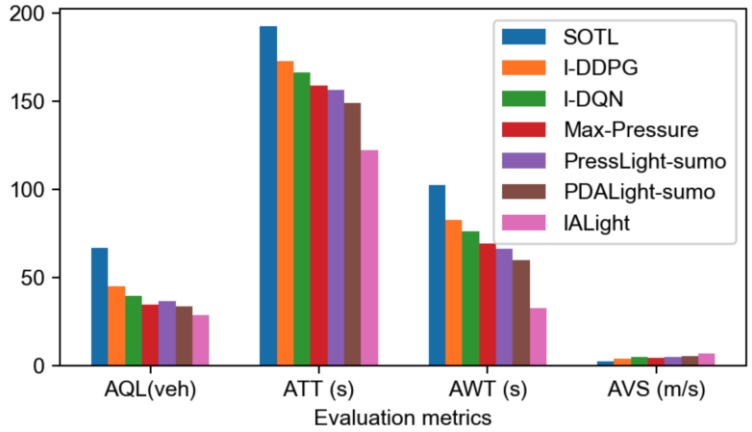


*Figure 16 – Evaluation metrics*

In addition to the overall control effect of the whole arterial, our proposed IALight can also focus on and improve the operation efficiency of the critical node with the highest importance in the arterial network. It is widely known that the critical intersection is the main factor and the bottlenecks that affect the operational efficiency of the arterial. If the critical intersection is not managed effectively, it can increase the probability and spread of congestion, and even may leading to unexpected large-scale congestion. Therefore, we examined the metrics of the second intersection of the Yangzhuang arterial, which is set as the critical node. As shown in *Figure 17, 18 and 19*, the proposed method could also improve the queue length, average speed and delay better than other baseline methods on the intersection with the highest importance.

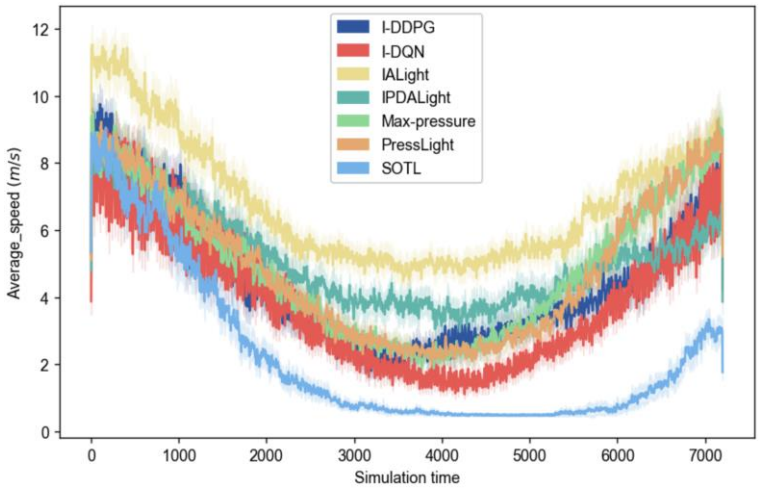


*Figure 17 – Average speed comparison on Intersection 2*

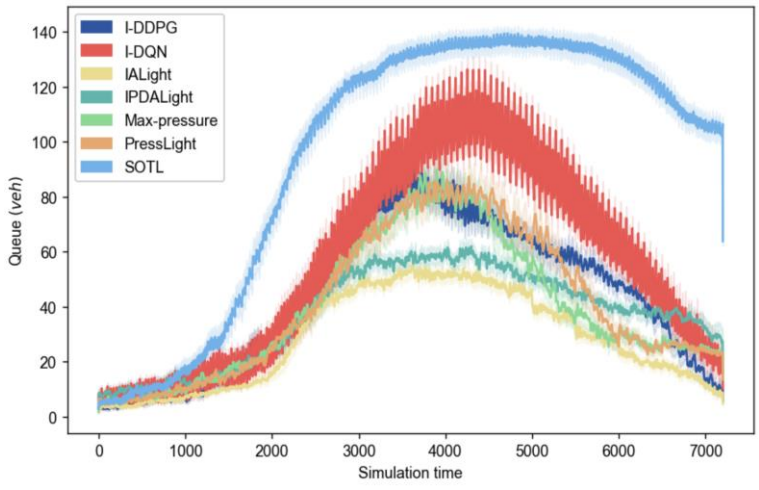


*Figure 18 – Queue comparison on Intersection 2*

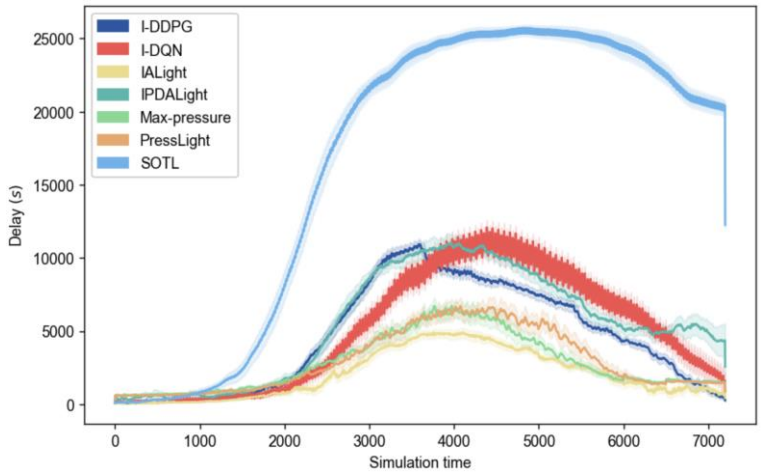


*Figure 19 – Delay comparison on Intersection 2*

We further examined the results of each intersection along the Yangzhuang arterial under the proposed IALight control, as shown in *Table 5*. It can be seen that as the critical node, intersection 2 manages to maintain effective traffic control even under the heaviest traffic flow. Notably, intersection 4 serves as a pedestrian crossing, with fewer stages, resulting in comparatively shorter waiting times and delays for motor vehicles.

*Table 5 – Metrics of each intersection of Yangzhuang Street*

| Intersection | AQL [veh] | AWT [s] | AVS [m/s] |
|---|---|---|---|
| 1 | 32.115 | 15.938 | 6.844 |
| **2** | 28.618 | 14.025 | 7.861 |
| 3 | 29.031 | 15.182 | 7.754 |
| 4 | 10.802 | 9.605 | 7.793 |
| 5 | 31.096 | 15.449 | 6.905 |

## 6. CONCLUSION

In this paper, an intersection importance-aware MARL-based TSC method is proposed for arterial traffic coordination control. In agent formulation, we propose a novel traffic intensity considering both stopped and moving vehicle state information to reflect the traffic demand efficiently. To enhance the practicality of our method, we employed a DDPG agent to optimise phase green time under a fixed cyclical phase sequence. Finally, an intersection importance-aware MARL model based on the MADDPG named IALight is proposed to improve the global arterial operation performance measures. Our IALight model is evaluated on the SUMO simulation software under a synthetic and a real-world road network in Beijing. The simulation results show that the proposed method outperforms the traditional and DRL-based baselines and improves the arterial operation effectively.

It is worth noting that we conducted our work based on the importance of each intersection is known and did not investigate how to evaluate the importance. The importance evaluation is quite an important work for traffic management, it is determined to be not only related to the static structure of the road network but also closely tied to the operational status of the traffic flow. These will be taken into account in our future work to provide more detailed information for arterial or network-wide cooperative control.

## ACKNOWLEDGEMENTS

This work was funded by the Key Research Project of the Beijing Polytechnic College [grant number BGY2024 KY-20Z], the Key Research Project of the Beijing Polytechnic College [grant number BGY2019 KY-05ZY].

## REFERENCES


[1] Mao F, Li Z, Li L. A comparison of deep reinforcement learning models for isolated traffic signal control. *IEEE Intelligent Transportation Systems Magazine*. 2023;15(1):169-180. DOI: 10.1109/MITS.2022.3144797.

[2] Haddad TA, Hedjazi D, Aouag S. A deep reinforcement learning-based cooperative approach for multi-intersection traffic signal control. *Engineering Applications of Artificial Intelligence*. 2022;114:105019. DOI: 10.1016/j.engappai.2022.105019.

[3] Wang T, Cao J, Hussain A. Adaptive traffic signal control for large-scale scenario with cooperative group-based multi-agent reinforcement learning. *Transportation Research Part C: Emerging Technologies*. 2021;125:103046. DOI: 10.1016/j.trc.2021.103046.

[4] Liu J, Zhang H, Fu Z, Wang Y. Learning scalable multi-agent coordination by spatial differentiation for traffic signal control. *Engineering Applications of Artificial Intelligence*. 2021;100:104165. DOI: 10.1016/j.engappai.2021.104165.

[5] Zhao W, et al. IPDALight: Intensity-and phase duration-aware traffic signal control based on reinforcement learning. *Journal of Systems Architecture*. 2022;123:102374. DOI: 10.1016/j.sysarc.2021.102374.

[6] Chandan K, Seco AM, Silva AB. Real-time traffic signal control for isolated intersection, using car-following logic under connected vehicle environment [J]. *Transportation Research Procedia*. 2017;25:1610-1625. DOI: 10.1016/j.trpro.2017.05.207.

[7] Kustija J. SCATS (Sydney Coordinated Adaptive Traffic System) as A solution to overcome traffic congestion in big cities. *International Journal of Research and Applied Technology (INJURATECH)*. 2023;3(1):1-14. DOI: 10.34010/injuratech.v3i1.7875.

[8] Studer L, Ketabdari M, Marchionni G. Analysis of adaptive traffic control systems design of a decision support system for better choices. *Journal of Civil & Environmental Engineering*. 2015; 5(6): 1-10.

[9] Li L, Lv Y, Wang FY. Traffic signal timing via deep reinforcement learning. *IEEE/CAA Journal of Automatica Sinica*. 2016;3:247–254. DOI: 10.1109/JAS.2016.7508798.

[10] Genders W, Razavi S. Using a deep reinforcement learning agent for traffic signal control. *arXiv preprint arXiv*:1611.01142 2016. DOI: 10.48550/arXiv.1611.01142.

[11] Liang X, Du X, Wang G, Han Z. A deep reinforcement learning network for traffic light cycle control. *IEEE Transactions on Vehicular Technology*. 2019;68:1243–1253. DOI: 10.1109/TVT.2018.2890726.

[12] Zhang L, Deng J. Data might be enough: Bridge real-world traffic signal control using offline reinforcement learning. *arXiv preprint arXiv*:2303;10828 2023. DOI: 10.48550/arXiv.2303.10828.

[13] Wei H, et al. Presslight: Learning max pressure control to coordinate traffic signals in arterial network. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2019:1290–1298. DOI: 10.1145/3292500.3330949.

[14] Li S. Multi-agent deep deterministic policy gradient for traffic signal control on urban road network. 2020 *IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*. IEEE. 2020:896–900. DOI: 10.1109/AEECA49918.2020.9213523.

[15] Ma D, Chen X, Wu X, Jin S. Mixed-coordinated decision-making method for arterial signals based on reinforcement learning. *Journal of Transportation Systems Engineering and Information Technology*. 2022;22:145. DOI: 10.16097/j.cnki.1009-6744.2022.02.014.

[16] Wei H, et al. Colight: Learning network-level cooperation for traffic signal control. *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2019;1913–1922. DOI: 10.1145/3357384.3357902.

[17] Chen C, et al. Toward a thousand lights: Decentralized deep reinforcement learning for large-scale traffic signal control. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2020;34(4):3414–3421. DOI: 10.1609/aaai.v34i04.5744.

[18] Xu M, et al. Discovery of critical nodes in road networks through mining from vehicle trajectories. *IEEE Transactions on Intelligent Transportation Systems*. 2018;20:583–593. DOI: 10.1109/TITS.2018.2817282.

[19] Xu M, et al. Networkwide traffic signal control based on the discovery of critical nodes and deep reinforcement learning. *Journal of Intelligent Transportation Systems*. 2020;24:1–10. DOI: 10.1080/15472450.2018.1527694.

[20] Zhang W, et al. Distributed signal control of arterial corridors using multi-agent deep reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*. 2023;24(1):178-190. DOI: 10.1109/TITS.2022.3216203.

[21] Zeng J, et al. Halight: Hierarchical deep reinforcement learning for cooperative arterial traffic signal control with cycle strategy. *IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. 2022;479-485. DOI: 10.1109/ITSC55140.2022.9921819.

[22] Fang Z, et al. MonitorLight: Reinforcement learning-based traffic signal control using mixed pressure monitoring. *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 2022;478–487. DOI: 10.1145/3511808.3557400.

[23] Yang H, et al. Deep reinforcement learning based strategy for optimizing phase splits in traffic signal control. *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. 2022;2329–2334. DOI:10.1109/ITSC55140.2022.9922531.

[24] Ibrokhimov B, Kim YJ, Kang S. Biased pressure: Cyclic reinforcement learning model for intelligent traffic signal control. *Sensors*. 2022;22:2818. DOI: 10.3390/s22072818.

[25] Barman S, Levin MW. Performance evaluation of modified cyclic max-pressure controlled intersections in realistic corridors. *Transportation Research Record*. 2022;2676:110–128. DOI: 10.1177/03611981211072807.

[26] Huang X, et al. Traffic node importance evaluation based on clustering in represented transportation networks. *IEEE Transactions on Intelligent Transportation Systems*. 2022;23:16622–16631. DOI: 10.1109/TITS.2022.3163756.

[27] Liu J, Li X, Dong J. A survey on network node ranking algorithms: Representative methods, extensions, and applications. *Science China Technological Sciences*. 2021;64:451–461. DOI: 10.1007/s11431-020-1683-2.

[28] Lowe R, et al. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems*. 2017;30.

[29] Monga R, Mehta D. Sumo (Simulation of Urban Mobility) and OSM (Open Street Map) Implementation. *2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART)*. *IEEE*. 2022; 534–538. DOI: 10.1109/SMART55829.2022.10046720.

[30] Ferreira M, et al. Self-organized traffic control. *Proceedings of the seventh ACM international workshop on VehiculAr InterNETworking*. 2010;85–90. DOI: 10.1145/1860058.1860077.

[31] Varaiya P. Max pressure control of a network of signalized intersections. *Transportation Research Part C: Emerging Technologies*. 2013;36:177–195. DOI: 10.1016/j.trc.2013.08.014.

[32] Liu P, et al. Traffic signal timing optimization based on intersection importance in vehicle-road collaboration. *Machine Learning for Cyber Security*. 2023(14541). DOI: 10.1007/978-981-97-2458-1_6.

[33] Huang X, et al. Traffic node importance evaluation based on clustering in represented transportation networks. *IEEE Transactions on Intelligent Transportation Systems*. 2022;23(9): 16622-16631. DOI: 10.1109/TITS.2022.3163756.

魏路，张小燕，樊利军，高磊，杨建

**IALight：基于重要度感知的多智能体强化学习干线交通协同控制**

**摘要：**
在城市交通管理中，多路口协同控制对于优化干线或网络交通运行效率具有重要意义。在干线协同控制方面，多智能体强化学习（MARL）被认为是一种高效的解决方案并具有良好的效果。然而，大多数现有的基于 MARL 的方法对各个路口一视同仁，忽视了每个路口的重要性差异，例如不同路口是否具有交通流量高、连接多条主要道路、作为高速公路或商业区的入口或出口点等特性。同时，如何提高强化学习交通控制方法的效率和实用性仍然是一项具有挑战性的工作。为了解决这些问题，本文提出了一种新颖的基于重要性感知的 MARL 方法用于交通优化控制，并命名为 IALight。首先，我们引入了归一化的交通压力，以确保交叉口强化学习智能体的状态和奖励设计能够准确反映交通流的状态；其次，设计了一个奖励调节模块，可根据路口的重要度对奖励进行修正和调节。为了提高现实所提方法的实用性和安全性，我们设计了一种固定相序模式下的绿灯时长优化策略。基于虚拟路网和实际路网仿真建模的交通场景中的综合实验表明，所提出的 IALight 相比传统交通控制方法和基于深度强化学习交通控制等基准方法，平均车辆行程时间分别降低了 20.41%和 17.88%。

**关键词：**
交通信号控制；交叉口重要度；多智能体强化学习；干线协同控制。